

תכנות מונחה עצמים תש"ף - סמסטר ק'

**תרגיל בית מס' 2**

העמסת אופרטורים

בדומה לתרגיל 1 - בתרגיל זה נעסוק בשברים של פולינומים בעל משתנה יחיד x.   
פולינום הוא סכום של מונומים בעל משתנה יחיד x.

דוגמא:

פולינום לדוגמא 1 +*x*2 − 2 · *x*  ·p(x)=3 .

שבר לדוגמא

כל פולינום ייוצג על ידי עצם במחלקה Polynomial. לכל עצם מסוג פולינום מוגדרת הדרגה המקסימלית שלו. הפולינום שבדוגמא יכול להיות מיוצג על ידי עצם מדרגה 2 ומעלה.

שבר מיוצג על ידי שני פולינומיאלים – המכנה והמונה.

יש להוסיף לתרגיל 1 עבור אוביקט Polynomial העמסת אופרטורים +,-,\*,"=" באופן הגיוני. כמו כן יש לממש שני אופרטורים [ ] – עם ובלי const - כך שיתאפשר לקרוא ולשנות מקדם של פולינום.

על מנת לבדוק את התוכנית שלך יש להעתיק את התכנית הראשית המופיעה מטה לקובץ בשם Main.cpp והפלט המתקבל צריך להיות זהה לפלט שרשום בהמשך.

דרישות נוספות:

○ המנעו משכפול קוד וקוד מיותר.

○ הדרגה של פולינום לא ניתנת לשינוי לאחר יצירתו, יש להגדיר אותה כ - const.

○ יש להשתמש ב const ו references בכל מקום שראוי להשתמש בהם.

**אין להשתמש במחלקה מסוג מונום. המחלקה Polynomial לא מכילה אף אוביקט ששיך למחלקה אחרת.**

בנוסף המחלקה Polynomial תשמור את הדרגה המקסימלית של כל הפולינומים שהוגדרו. ניתן יהיה להדפיס ערך זה בכל נקודת זמן.

**הוראות הגשה**

* יש להגיש קובץ ZIP בשם
* HW2\_123456789\_987654321.zip

(יש להחליף את המספרים עם מספרי ת.ז. של המגישים.)

* על הקובץ להכיל את כל קבצי ה cpp ו h שכתבתם.
* שימו לב: הקובץ לא צריך להכיל את התיקיה שבה הקבצים נמצאים, אלא רק את הקבצים עצמם.
* רק אחד מהשותפים לתרגיל יגיש אותו ב MOODLE.

**תכנית ראשית**

// HWCPP1.cpp : Defines the entry point for the console application.

//

#include <iostream>

#include "Polynomial.h"

#include "rational.h"

void testPolynomial()

{

cout << "----- start testPolynomial ---------" << endl;

cout << "test0 :" << Polynomial::getMaxDegree() <<endl;

Polynomial p1;

cout << "test1: " <<p1;

Polynomial p2(16);

p2.setCoeff(2, 0.4);

cout << "test2: " << p2 << "degree=" << p2.getDegree(false)<<endl;

double c[5] = { 0.1, 0.2, 0 ,11.5, 1.3};

Polynomial p3(c,4);

cout << "test3: " << p3;

p2.setCoeff(2, 0.0);

cout << "test4: " << p2;

p2 = (p1 + 2.5\*p3 -0.9\*p3)\*p3;

p2[5] = 1;

cout << "test5 " << p2 << endl << "(2\*p2)[6] = " << (2\*p2)[6] <<endl;

cout << "test6 :maxDegree = " << Polynomial::getMaxDegree() << endl;

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<< endl;

}

void testRational() {

cout << "----- start testRational ---------" << endl;

cout << "test7" << endl;

rational r1;

r1.print();

cout << r1;

double c[] = { 0,2,2,3,4,5 };

Polynomial p1(c, 5);

c[0] = 1.2;

Polynomial p2(c, 3);

rational r2(p1, p2);

cout << "test8" << endl;

r2.print();

cout << endl<< r2<<endl;

cout << "test9" << endl;

cout << "nom = " << r2.getNom();

cout << "denom = " << r2.getDenom();

rational r3 = r2\*r2 + r2;

cout << "test10 : r3 = " << endl << r3 << endl;

double d[21] = {0};

d[0] = 1;

Polynomial p3(d, 20);

rational r4(p3, p3);

r4 = (r2 + r4)\*r2;

cout << "test11 : r4 = " << endl << r4 << endl;

cout << "test12 :maxDegree = " << Polynomial::getMaxDegree() << endl;

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

}

int main()

{

testPolynomial();

testRational();

}

----- start testPolynomial ---------

test0 :0

test1: polynomial = 0

test2: polynomial = 0+(0)\*X^1+(0.4)\*X^2

degree=16

test3: polynomial = 0.1+(0.2)\*X^1+(0)\*X^2+(11.5)\*X^3+(1.3)\*X^4

test4: polynomial = 0

test5 polynomial = 0.016+(0.064)\*X^1+(0.064)\*X^2+(3.68)\*X^3+(7.776)\*X^4+(1)\*X^5+(211.6)\*X^6+(47.84)\*X^7+(2.704)\*X^8

(2\*p2)[6] = 423.2

test6 :maxDegree = 8

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

----- start testRational ---------

test7

polynomial = 0

--------------------------

polynomial = 1

polynomial = 0

--------------------------

polynomial = 1

test8

polynomial = 0+(2)\*X^1+(2)\*X^2+(3)\*X^3+(4)\*X^4+(5)\*X^5

--------------------------

polynomial = 1.2+(2)\*X^1+(2)\*X^2+(3)\*X^3

polynomial = 0+(2)\*X^1+(2)\*X^2+(3)\*X^3+(4)\*X^4+(5)\*X^5

--------------------------

polynomial = 1.2+(2)\*X^1+(2)\*X^2+(3)\*X^3

test9

nom = polynomial = 0+(2)\*X^1+(2)\*X^2+(3)\*X^3+(4)\*X^4+(5)\*X^5

denom = polynomial = 1.2+(2)\*X^1+(2)\*X^2+(3)\*X^3

test10 : r3 =

polynomial = 0+(2.88)\*X^1+(17.28)\*X^2+(49.12)\*X^3+(111.36)\*X^4+(208.8)\*X^5+(326.8)\*X^6+(441.6)\*X^7+(511.2)\*X^8+(518)\*X^9+(430)\*X^10+(313)\*X^11+(170)\*X^12+(75)\*X^13

--------------------------

polynomial = 1.728+(8.64)\*X^1+(23.04)\*X^2+(49.76)\*X^3+(81.6)\*X^4+(103.2)\*X^5+(112.4)\*X^6+(90)\*X^7+(54)\*X^8+(27)\*X^9

test11 : r4 =

polynomial = 0+(2.4)\*X^1+(10.4)\*X^2+(19.6)\*X^3+(36.8)\*X^4+(54)\*X^5+(72)\*X^6+(66)\*X^7+(61)\*X^8+(40)\*X^9+(25)\*X^10

--------------------------

polynomial = 1.44+(4.8)\*X^1+(8.8)\*X^2+(15.2)\*X^3+(16)\*X^4+(12)\*X^5+(9)\*X^6

test12 :maxDegree = 13

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

בהצלחה !!